Modeling

Aleksei Sorokin, asorokin@hawk.iit.edu, A20394300

Load Data

```
df <- read.csv('data/all_coaches.csv')
data_dict <- list(
              N='Name',
              GR='Games Relative',
              WP='Win Loss Percentage',
              PGR='Playoff Games Relative',
              PWP='Playoff Win Percentage',
              CC='Conference Championships',
              C='Championships',
              HOF='Hall of Fame',
              S='Sport')
names(df) <- names(data_dict)
head(df)
```

```
##              N       GR        WP       PGR   PWP CC C HOF        S
## 1     AJ Hinch 6.308642 0.5580000 4.5454545 0.560  2 1   0 baseball
## 2  Aaron Boone 2.000000 0.6270000 1.2727273 0.500  0 0   0 baseball
## 3 Aaron Kromer 0.375000 0.3330000 0.0000000 0.000  0 0   0 football
## 4   Abe Gibron 2.625000 0.2740000 0.0000000 0.000  0 0   0 football
## 5    Adam Gase 4.000000 0.4690000 0.3333333 0.000  0 0   0 football
## 6   Adam Oates 1.585366 0.5752212 0.4375000 0.429  0 0   0   hockey
```

Split into train test datasets

```
set.seed(7)
train_frac <- 4/5
df_HOF_1 <- df[df$HOF==1,]
df_HOF_0 <- df[df$HOF==0,]
HOF_1_train <- sample(1:nrow(df_HOF_1),floor(nrow(df_HOF_1)*train_frac))
HOF_0_train <- sample(1:nrow(df_HOF_0),floor(nrow(df_HOF_0)*train_frac))
df_train <- rbind(df_HOF_1[HOF_1_train,],df_HOF_0[HOF_0_train,])
df_test <- rbind(df_HOF_1[-HOF_1_train,],df_HOF_0[-HOF_0_train,])
l1 <- sprintf('Train Fraction: %.2f',train_frac)
l2 <- sprintf('Hall of Fame Coaches: %d. (Train %d , Test %d)',
        nrow(df_HOF_1),length(HOF_1_train),nrow(df_HOF_1)-length(HOF_1_train))
l3 <- sprintf('Non Hall of Fame Coaches: %d. (Train %d , Test %d)',
        nrow(df_HOF_0),length(HOF_0_train),nrow(df_HOF_0)-length(HOF_0_train))
l4 <- sprintf('Overall: (Train %d , Test %d)',nrow(df_train),nrow(df_test))
cat(sprintf('%s\n%s\n%s\n%s\n',l1,l2,l3,l4))
```

```
## Train Fraction: 0.80
## Hall of Fame Coaches: 256. (Train 204 , Test 52)
## Non Hall of Fame Coaches: 1664. (Train 1331 , Test 333)
## Overall: (Train 1535 , Test 385)
```

Standard Logistic Regression Model

```
model_1 <- glm(HOF ~ GR + WP + PGR + PWP + CC + C + S,
               data=df_train,
               family="binomial")
summary(model_1)
```

```
##
## Call:
## glm(formula = HOF ~ GR + WP + PGR + PWP + CC + C + S, family = "binomial",
##     data = df_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.2546  -0.4892  -0.4029  -0.2870   2.7290
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.84044    0.32502  -8.739  < 2e-16 ***
## GR           0.07033    0.02675   2.629  0.00857 **
## WP           0.60457    0.63546   0.951  0.34140
## PGR         -0.15829    0.06935  -2.282  0.02246 *
## PWP         -0.97670    0.50915  -1.918  0.05508 .
## CC           0.73104    0.16582   4.409 1.04e-05 ***
## C            0.36571    0.21871   1.672  0.09450 .
## Sbasketball -0.69091    0.34668  -1.993  0.04627 *
## Sfootball    0.15983    0.22378   0.714  0.47509
## Shockey      1.09480    0.23838   4.593 4.38e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1203.01  on 1534  degrees of freedom
## Residual deviance:  990.98  on 1525  degrees of freedom
## AIC: 1011
##
## Number of Fisher Scoring iterations: 5
```

```
df_test$yHat_1 <- predict(model_1, newdata=df_test, type="response")
```

Logistic Mixed Model

```
library(lme4)
model_2 <- lmer(HOF ~ GR + WP + PGR + PWP + CC + C + (1|S),
                data = df_train)
summary(model_2)
```

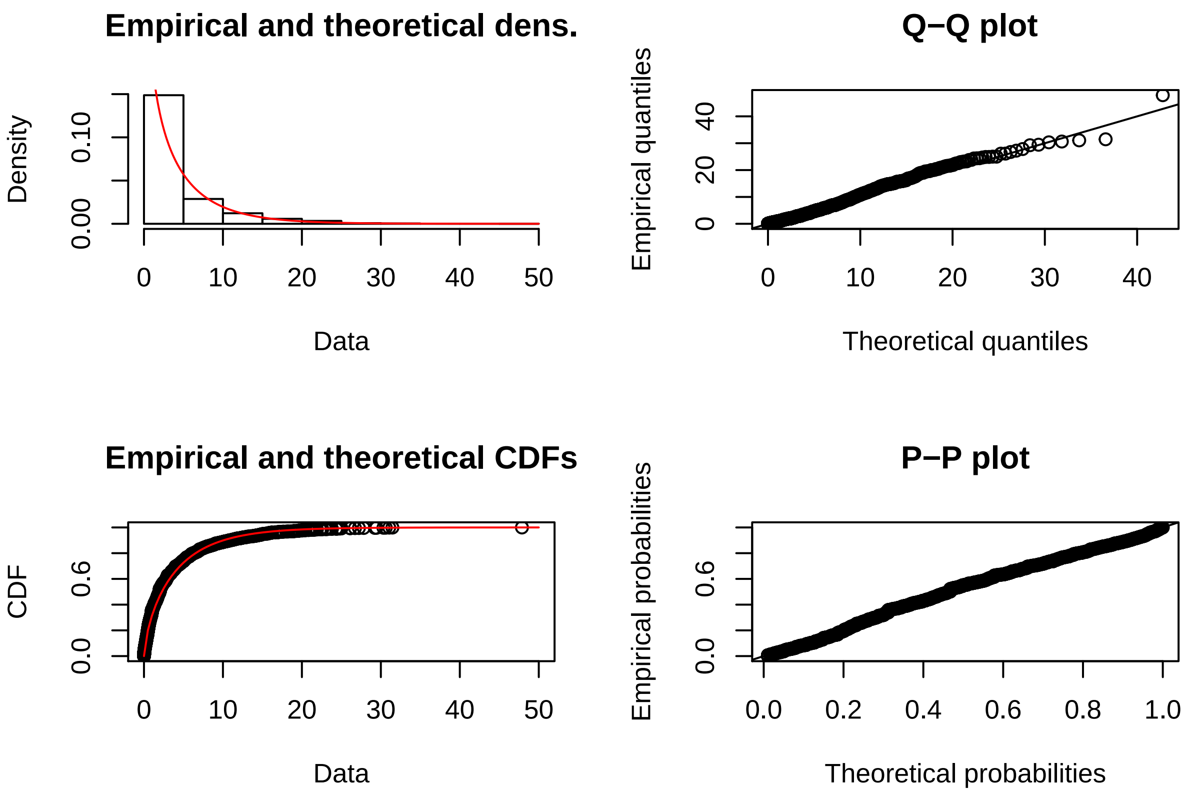
```
## Linear mixed model fit by REML ['lmerMod']
## Formula: HOF ~ GR + WP + PGR + PWP + CC + C + (1 | S)
##    Data: df_train
##
## REML criterion at convergence: 795.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5487 -0.4286 -0.2610 -0.1408  3.2702
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  S        (Intercept) 0.00429  0.0655
##  Residual             0.09513  0.3084
## Number of obs: 1535, groups:  S, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.066260   0.040566   1.633
## GR           0.008219   0.002676   3.072
## WP           0.054907   0.056677   0.969
## PGR         -0.018178   0.006785  -2.679
## PWP         -0.088286   0.048663  -1.814
## CC           0.103399   0.017237   5.999
## C            0.031050   0.021654   1.434
##
## Correlation of Fixed Effects:
##     (Intr) GR     WP     PGR    PWP    CC
## GR  -0.077
## WP  -0.524 -0.040
## PGR  0.064 -0.544 -0.057
## PWP  0.063 -0.264 -0.275 -0.141
## CC   0.047 -0.316 -0.009 -0.042  0.006
## C   -0.019  0.249  0.005 -0.192 -0.079 -0.817
```

```
df_test$yHat_2 <- predict(model_2,df_test, type="response")
```

Bayesian Logistic Mixed Model

Games Relative (GR)

```
library(fitdistrplus)
fit_gr <- fitdist(df$GR,'gamma')
plot(fit_gr)
```



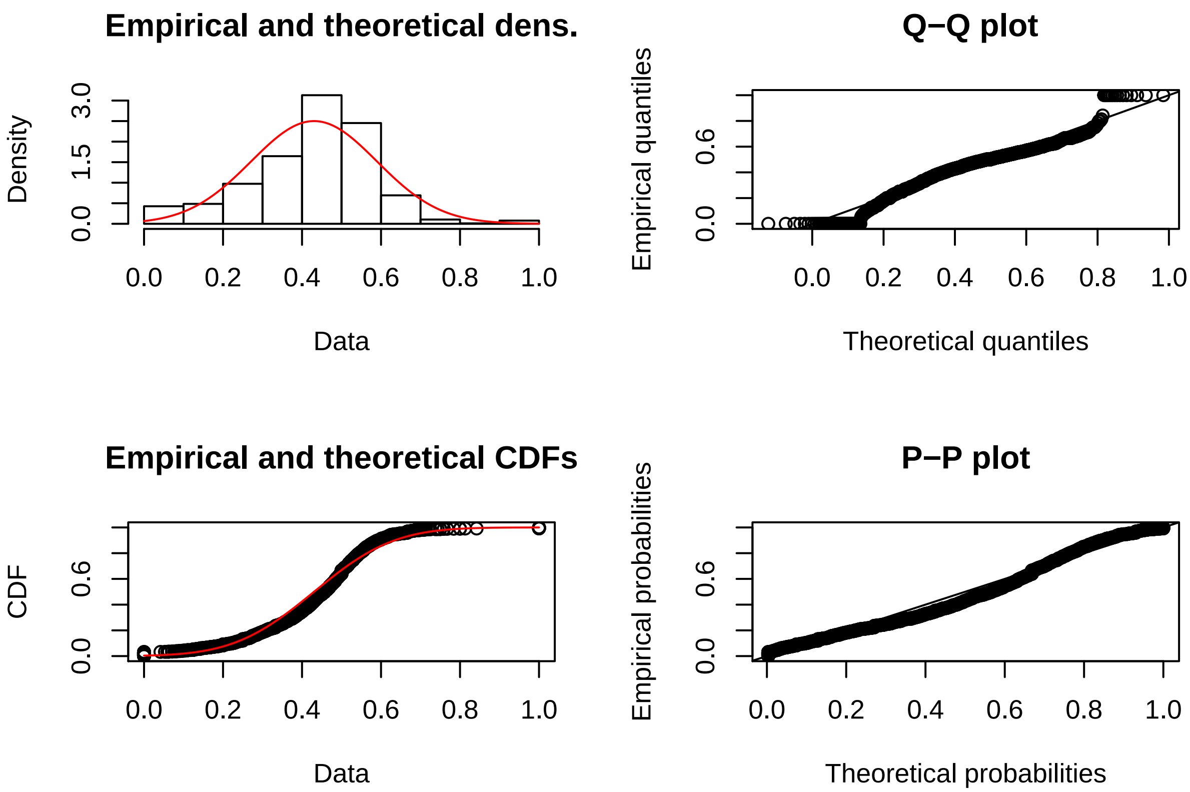

```
summary(fit_gr)
```

```
## Fitting of the distribution ' gamma ' by maximum likelihood
## Parameters :
##        estimate  Std. Error
## shape 0.6763870 0.018540327
## rate  0.1702245 0.006639272
## Loglikelihood:  -4455.053   AIC:  8914.106   BIC:  8925.226
## Correlation matrix:
##           shape      rate
## shape 1.0000000 0.7027467
## rate  0.7027467 1.0000000
```

```
# --> GR ~ Gamma( .676 , .170 )
```

Win-Loss Percentage (WP)

```
fit_wp <- fitdist(df$WP,'norm')
plot(fit_wp)
```



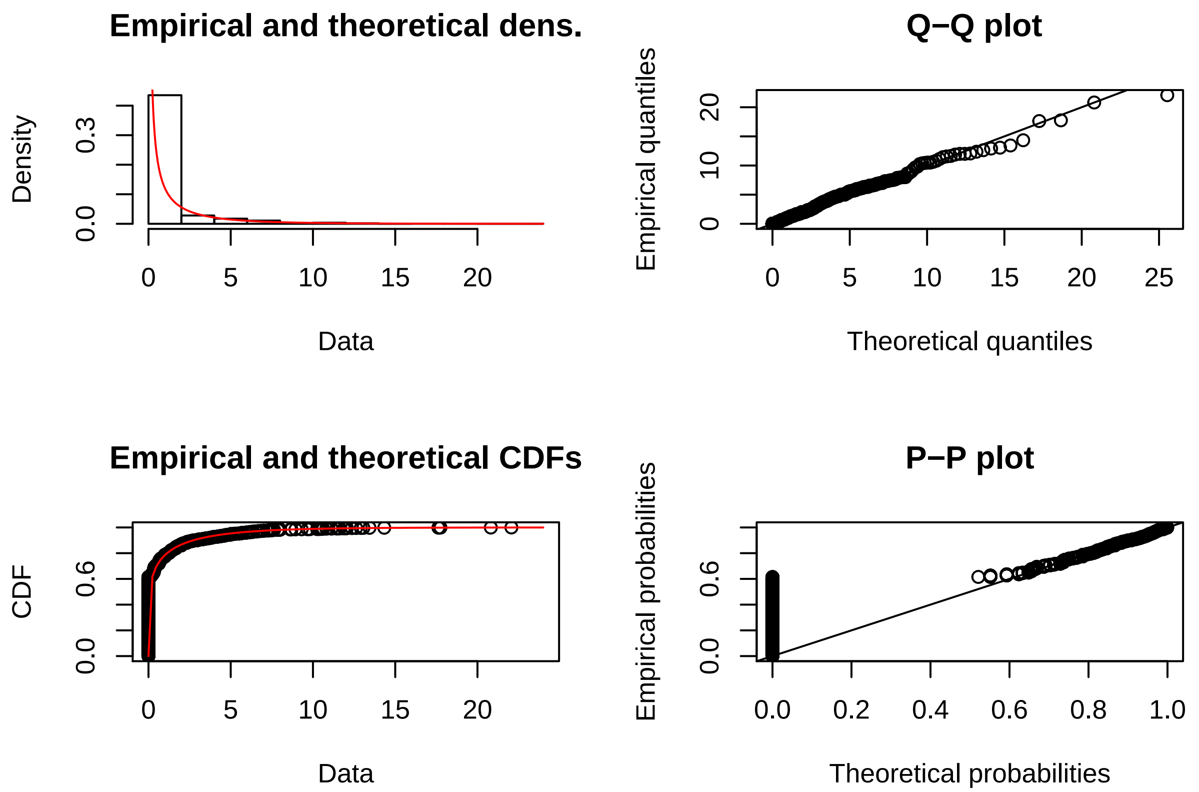

```r
summary(fit_wp)
```

```
## Fitting of the distribution ' norm ' by maximum likelihood
## Parameters :
##       estimate  Std. Error
## mean 0.4304723 0.003641454
## sd   0.1595605 0.002574441
## Loglikelihood:  799.4757   AIC:  -1594.951   BIC:  -1583.831
## Correlation matrix:
##      mean sd
## mean    1  0
## sd      0  1
```

```r
# --> WP ~ N( .430 , .160 )
```

Playoff Games Relative (PGR)

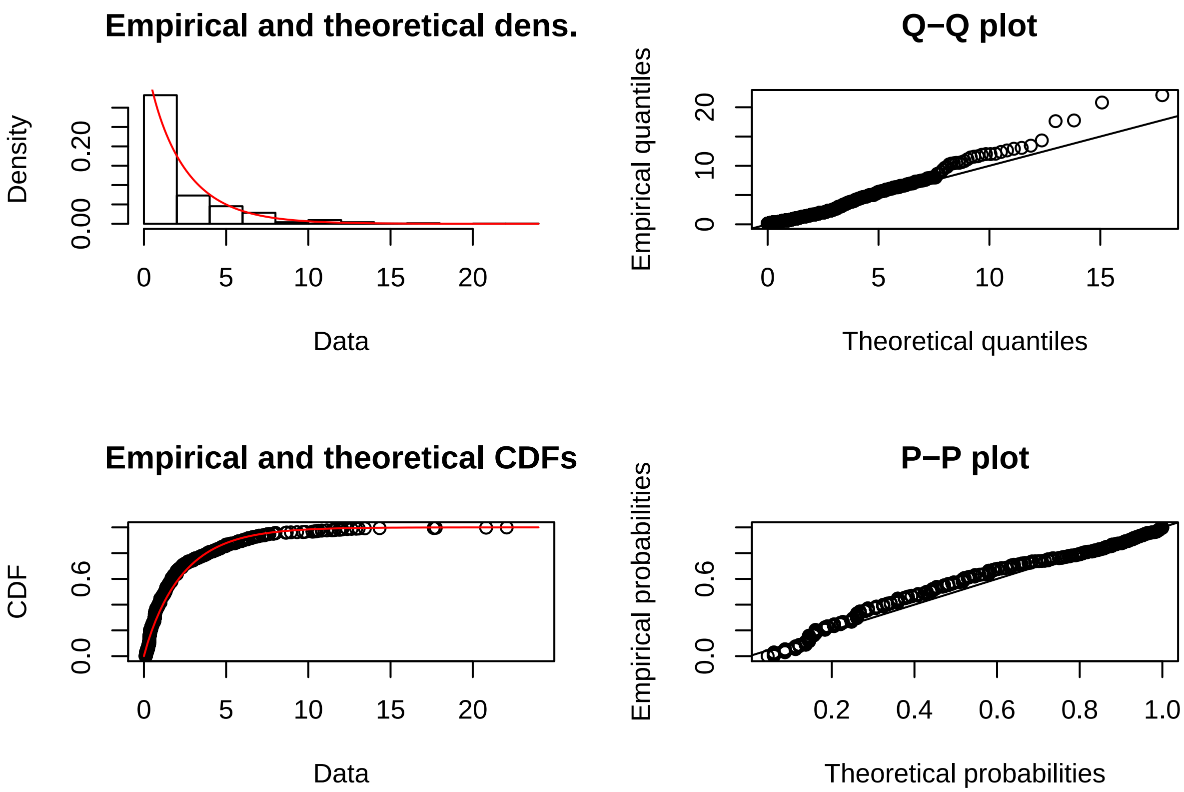
```r
# include coaches with 0 playoff games
fit_pgr <- fitdist(df$PGR,'gamma','mme')
plot(fit_pgr)
```




```r
summary(fit_pgr)
```

```
## Fitting of the distribution ' gamma ' by matching moments
## Parameters :
##        estimate
## shape 0.1824340
## rate  0.2021443
## Loglikelihood:  Inf   AIC:  -Inf   BIC:  -Inf
```

```r
# ony coaches with > 0 playoff games
df_pgr_gt0 <- df[df$PGR>0,]
fit_pgr_gt0 <- fitdist(df_pgr_gt0$PGR,'gamma')
plot(fit_pgr_gt0)
```



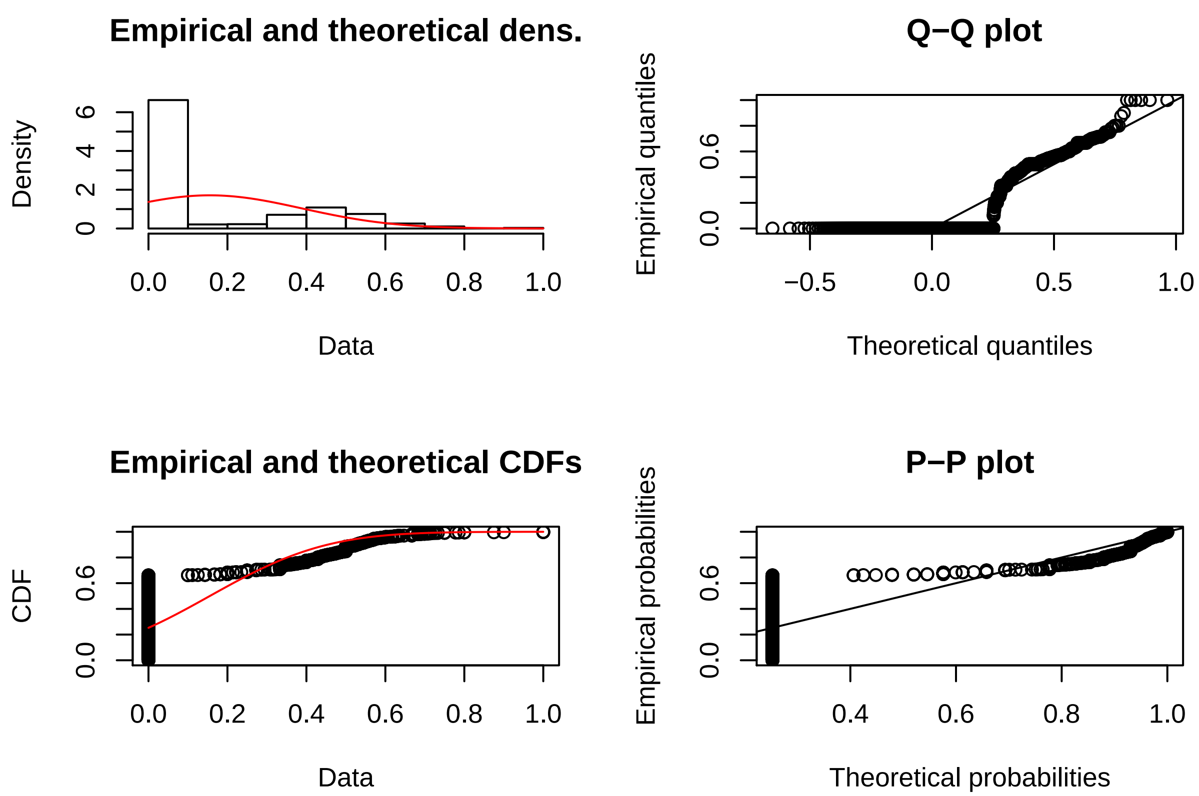

```r
summary(fit_pgr_gt0)
```

```
## Fitting of the distribution ' gamma ' by maximum likelihood
## Parameters :
##        estimate Std. Error
## shape 0.9402688 0.04283310
## rate  0.4010054 0.02377207
## Loglikelihood:  -1367.831   AIC:  2739.662   BIC:  2748.873
## Correlation matrix:
##           shape      rate
## shape 1.0000000 0.7684334
## rate  0.7684334 1.0000000
```

```r
# --> PGR ~ Gamma( .940 , .401 )
```

Playoff Win-Loss Percentage (PWP)

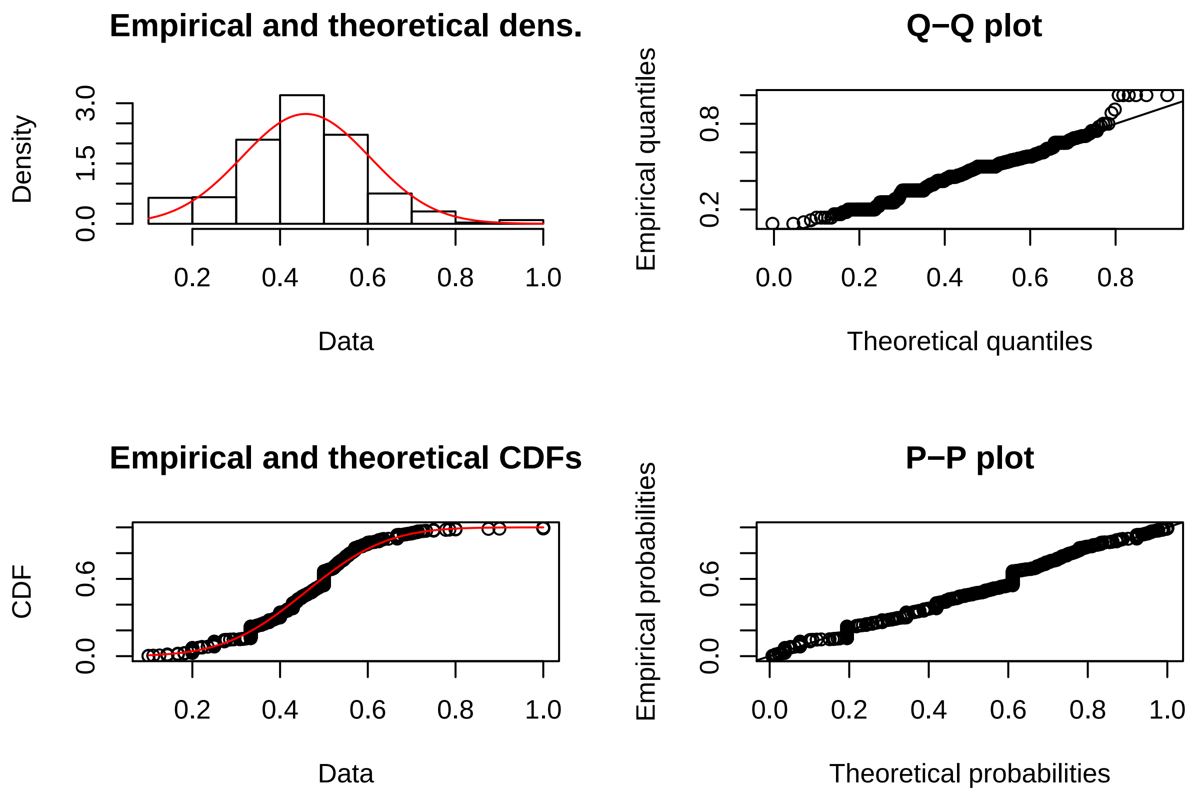
```r
# include coaches with 0% playoff win-loss-percentage
fit_pwp <- fitdist(df$PWP,'norm')
plot(fit_pwp)
```




```r
summary(fit_pwp)
```

```
## Fitting of the distribution ' norm ' by maximum likelihood
## Parameters :
##       estimate  Std. Error
## mean 0.1553094 0.005319974
## sd   0.2331096 0.003761478
## Loglikelihood:  71.6316   AIC:  -139.2632   BIC:  -128.143
## Correlation matrix:
##              mean           sd
## mean 1.000000e+00 1.635144e-12
## sd   1.635144e-12 1.000000e+00
```

```r
# only coaches with > 0% playoff win-loss percentage
df_pwp_gt0 <- df[df$PWP>0,]
fit_pwp_gt0 <- fitdist(df_pwp_gt0$PWP,'norm')
plot(fit_pwp_gt0)
```



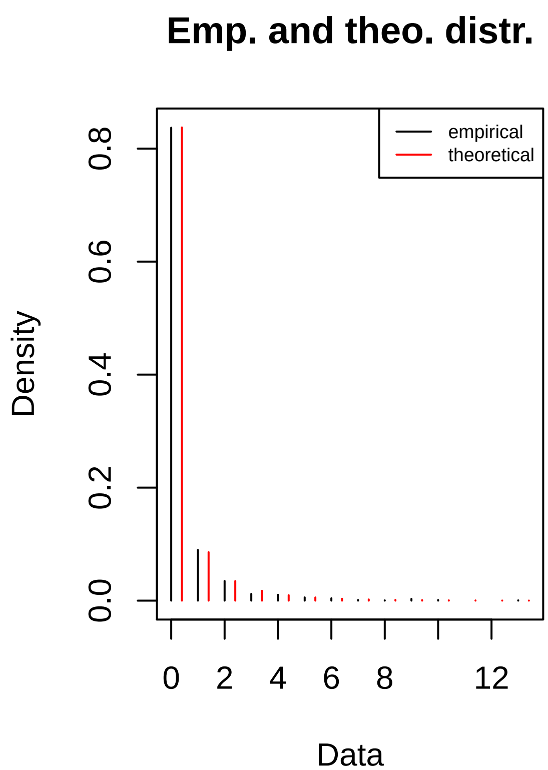

```r
summary(fit_pwp_gt0)
```

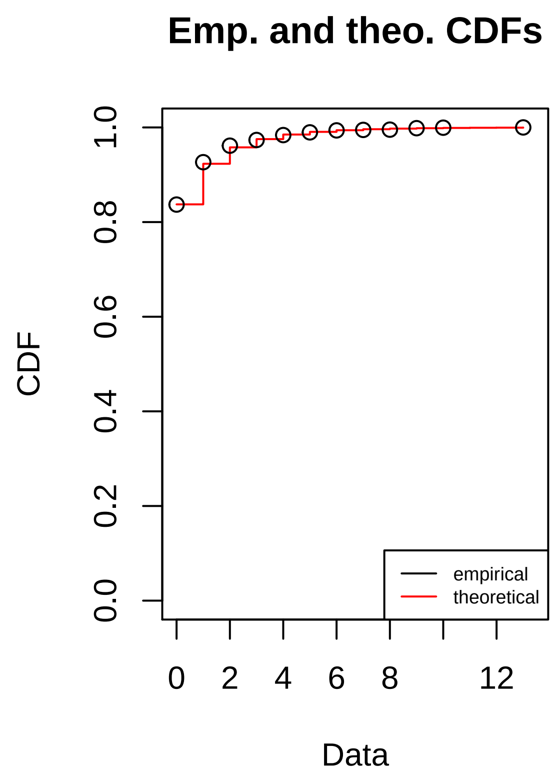
```
## Fitting of the distribution ' norm ' by maximum likelihood
## Parameters :
##        estimate  Std. Error
## mean 0.4587600 0.005724604
## sd   0.1459493 0.004047051
## Loglikelihood:  328.6122   AIC:  -653.2243   BIC:  -644.2704
## Correlation matrix:
##      mean sd
## mean    1  0
## sd      0  1
```

```r
# --> PWP ~ N( .459 , .146 )
```

Conference Championships (CC)

```r
fit_cc <- fitdist(df$CC,'nbinom')
plot(fit_cc)
```





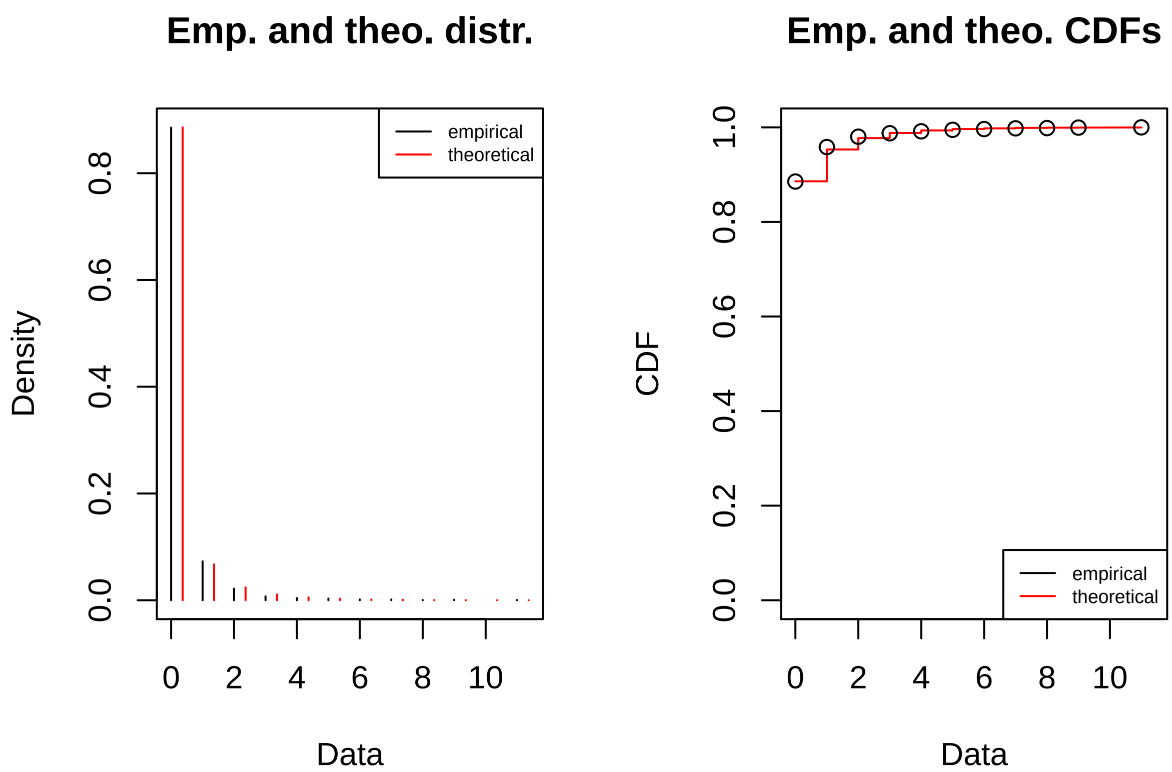

```r
summary(fit_cc)
```

```
## Fitting of the distribution ' nbinom ' by maximum likelihood
## Parameters :
##       estimate Std. Error
## size 0.1452930 0.01331281
## mu   0.3473364 0.02476231
## Loglikelihood:  -1306.131   AIC:  2616.262   BIC:  2627.383
## Correlation matrix:
##              size           mu
## size 1.0000000000 0.0001584868
## mu   0.0001584868 1.0000000000
```

```r
# --> CC ~ Neg-Binomial( .145 , .347 )
```

Championships (C)

```r
fit_c <- fitdist(df$C,'nbinom')
plot(fit_c)
```




```r
summary(fit_c)
```

```
## Fitting of the distribution ' nbinom ' by maximum likelihood
## Parameters :
##       estimate Std. Error
## size 0.1188586 0.01370002
## mu   0.2104236 0.01742436
## Loglikelihood:  -954.245   AIC:  1912.49   BIC:  1923.61
## Correlation matrix:
##               size            mu
## size  1.000000e+00 -2.062405e-05
## mu   -2.062405e-05  1.000000e+00
```

```r
# --> C ~ Neg-Binomial( .119 , 210 )
```

Fit Bayesian GLMM (Logistic) Model

```r
library(blme)
# Approximate fits by MLE estimation (as shown above)
# --> GR  ~ Gamma( .676 , .170 )
# --> WP  ~ N( .430 , .160 )
# --> PGR ~ Gamma( .940 , .401 )
# --> PWP ~ N( .459 , .146 )
# --> CC  ~ Neg-Binomial( .145 , .347 )
# --> C   ~ Neg-Binomial( .119 , 210 )
model_3 <- blmer(HOF ~ GR + WP + PGR + PWP + CC + C + (1|S),
                 data = df_train,
                 resid.prior = gamma,
                 fixef.prior = normal,
```

```r
                 cov.prior = invwishart)
summary(model_3)
```

```
## Cov prior  : S ~ invwishart(df = 0.002, scale = 0.102, posterior.scale = cov, common.scale = TRUE)
## Fixef prior: normal(sd = c(10, 2.5, ...), corr = c(0 ...), common.scale = TRUE)
## Resid prior: gamma(shape = 0, rate = 0, posterior.scale = var)
## Prior dev  : 17.1453
##
## Linear mixed model fit by REML ['blmerMod']
## Formula: HOF ~ GR + WP + PGR + PWP + CC + C + (1 | S)
##    Data: df_train
##
## REML criterion at convergence: 795.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5588 -0.4311 -0.2617 -0.1412  3.2809
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  S        (Intercept) 0.00452  0.06723
##  Residual             0.09457  0.30751
## Number of obs: 1535, groups:  S, 4
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.066405   0.041194   1.612
## GR           0.008224   0.002667   3.083
## WP           0.054459   0.056349   0.966
## PGR         -0.018198   0.006765  -2.690
## PWP         -0.088041   0.048419  -1.818
## CC           0.103409   0.017178   6.020
## C            0.031033   0.021578   1.438
##
## Correlation of Fixed Effects:
##     (Intr) GR     WP     PGR    PWP    CC
## GR  -0.075
## WP  -0.513 -0.041
## PGR  0.062 -0.544 -0.057
## PWP  0.061 -0.264 -0.274 -0.141
## CC   0.046 -0.316 -0.009 -0.043  0.005
## C   -0.019  0.249  0.005 -0.192 -0.079 -0.817
```

```r
df_test$yHat_3 <- predict(model_3, newdata=df_test, type="response")
```

Make predictions on test data and calculate metrics

```r
model_metrics <- function(df_test,yHat_col,model_name){
  yHat_b_col <- paste0(yHat_col,'b')
  df_test[,yHat_b_col] <-(df_test[[yHat_col]] >= .5)
  yHat_b <- df_test[[yHat_b_col]]
  hof <- df_test$HOF
```

```r
  tp <- sum((yHat_b==1 & hof==1))
  fp <- sum((yHat_b==1 & hof==0))
  fn <- sum((yHat_b==0 & hof==1))
  tn <- sum((yHat_b==0 & hof==0))
  accuracy <- (tp+tn)/(tp+tn+fp+fn)
  precision <- tp/(tp+fp)
  recall <- tp/(tp+fn)
  cat(sprintf('%s\n\tAccuracy: %.3f\n\tPrecision: %.3f\n\tRecall: %.3f\n',
              model_name,accuracy,precision,recall))
  return (df_test)}
```

Compare models

```r
df_test <- model_metrics(df_test,'yHat_1','Standard Logistic Regression')
```

```
## Standard Logistic Regression
##  Accuracy: 0.883
##  Precision: 0.706
##  Recall: 0.231
```

```r
df_test <- model_metrics(df_test,'yHat_2','Logistic Mixed Model')
```

```
## Logistic Mixed Model
##  Accuracy: 0.886
##  Precision: 0.900
##  Recall: 0.173
```

```r
df_test <- model_metrics(df_test,'yHat_3','Bayesian Mixed Model')
```

```
## Bayesian Mixed Model
##  Accuracy: 0.886
##  Precision: 0.900
##  Recall: 0.173
```

```r
df_test[df_test$HOF==1 & df_test$yHat_3b==F,]
```

```
##                        N         GR        WP       PGR   PWP CC C HOF
## 31       Alan Trammell  3.0185185 0.3820000 0.0000000 0.000  0 0   1
## 39         Alex Hannum 10.7682927 0.5330000 4.9375000 0.570  2 2   1
## 113          Bill Cook  1.4268293 0.3655914 0.0000000 0.000  0 0   1
## 114        Bill Cowher 15.0000000 0.6230000 7.0000000 0.571  2 1   1
## 156         Bill Terry  9.2345679 0.5550000 1.4545455 0.438  3 1   1
## 167       Billy Herman  2.8703704 0.4080000 0.0000000 0.000  0 0   1
## 374  Christy Mathewson  2.1296296 0.4820000 0.0000000 0.000  0 0   1
## 497       Denis Savard  1.7926829 0.4961832 0.0000000 0.000  0 0   1
## 536        Dit Clapper  2.8048780 0.5368421 1.5625000 0.320  0 0   1
## 587         Duke Keats  0.1341463 0.2222222 0.0000000 0.000  0 0   1
## 628       Eddie Gerard  5.1341463 0.4833333 1.3125000 0.500  1 1   1
## 655       Ernie Nevers  2.5625000 0.3080000 0.0000000 0.000  0 0   1
## 690     Frank Robinson 13.8333333 0.4750000 0.0000000 0.000  0 0   1
## 721     Gabby Hartnett  2.3641975 0.5360000 0.3636364 0.000  1 0   1
## 744       George Allen 10.5000000 0.7120000 3.0000000 0.222  1 0   1
## 745    George Armstrong  0.5731707 0.3953488 0.0000000 0.000  0 0   1
## 791       Greasy Neale  6.9375000 0.5940000 1.3333333 0.750  2 2   1
```

```
## 859        Herb Brooks  6.1707317 0.5000000  2.5000000 0.475   0 0    1
## 1028          Jim Ringo  1.4375000 0.1300000  0.0000000 0.000   0 0    1
## 1099      Joe Stydahar  3.0625000 0.4170000  1.0000000 0.667   1 1    1
## 1178      Johnny Evers  2.3148148 0.4840000  0.0000000 0.000   0 0    1
## 1249       Larry Brown 24.4146341 0.5480000 12.0625000 0.518   3 1    1
## 1277     Lenny Wilkens 30.3292683 0.5360000 11.1250000 0.449   2 1    1
## 1326      Lynn Patrick  5.4024390 0.4570637  2.5000000 0.413   0 0    1
## 1351         Marv Levy 15.9375000 0.5610000  6.3333333 0.579   4 0    1
## 1381   Mickey Cochrane  3.7037037 0.5820000  1.1818182 0.538   2 1    1
## 1455    Mordecai Brown  0.7037037 0.4420000  0.0000000 0.000   0 0    1
## 1463        Nap Lajoie  4.3209877 0.5500000  0.0000000 0.000   0 0    1
## 1488 Norm Van Brocklin 10.8125000 0.3980000  0.0000000 0.000   0 0    1
## 1506         Pat Burns 12.4268293 0.5866511  9.3125000 0.523   1 1    1
## 1513         Pat Quinn 17.0731707 0.5643564 11.4375000 0.514   0 0    1
## 1581 Rabbit Maranville  0.3271605 0.4340000  0.0000000 0.000   0 0    1
## 1643       Rick Pitino  5.0243902 0.4660000  0.8125000 0.462   0 0    1
## 1660   Roger Bresnahan  4.7777778 0.4320000  0.0000000 0.000   0 0    1
## 1711       Sammy Baugh  2.6250000 0.4290000  0.0000000 0.000   0 0    1
## 1737     Slick Leonard  5.4878049 0.4130000  0.0000000 0.000   0 0    1
## 1740  Sprague Cleghorn  0.5853659 0.4634146  0.2500000 0.500   0 0    1
## 1842      Tommy Gorman  3.9878049 0.5018868  1.5625000 0.600   2 2    1
## 1868           Ty Cobb  5.7592593 0.5190000  0.0000000 0.000   0 0    1
## 1897    Walter Johnson  5.9629630 0.5500000  0.0000000 0.000   0 0    1
## 1903     Wayne Gretzky  4.0000000 0.4703947  0.0000000 0.000   0 0    1
## 1909     Whitey Herzog 14.8703704 0.5320000  4.6363636 0.510   3 1    1
## 1918        Yogi Berra  5.7407407 0.5220000  1.7272727 0.474   2 0    1
##               S     yHat_1     yHat_2     yHat_3 yHat_1b yHat_2b yHat_3b
## 31     baseball 0.08338720 0.08882184 0.08877852   FALSE   FALSE   FALSE
## 39   basketball 0.16847313 0.25404221 0.25377449   FALSE   FALSE   FALSE
## 113      hockey 0.19399478 0.18747081 0.18781223   FALSE   FALSE   FALSE
## 114    football 0.25220837 0.27656087 0.27649277   FALSE   FALSE   FALSE
## 156    baseball 0.51134844 0.42554713 0.42555371    TRUE   FALSE   FALSE
## 167    baseball 0.08379313 0.08903180 0.08897601   FALSE   FALSE   FALSE
## 374    baseball 0.08322977 0.08700677 0.08691378   FALSE   FALSE   FALSE
## 497      hockey 0.21088899 0.19764817 0.19793308   FALSE   FALSE   FALSE
## 536      hockey 0.14385064 0.15154478 0.15186500   FALSE   FALSE   FALSE
## 587      hockey 0.16772254 0.16897427 0.16937286   FALSE   FALSE   FALSE
## 628      hockey 0.33366090 0.29085292 0.29125256   FALSE   FALSE   FALSE
## 655    football 0.08995583 0.09685299 0.09684961   FALSE   FALSE   FALSE
## 690    baseball 0.17075000 0.18281518 0.18278934   FALSE   FALSE   FALSE
## 721    baseball 0.15754276 0.18868850 0.18857588   FALSE   FALSE   FALSE
## 744    football 0.18659288 0.21353844 0.21340404   FALSE   FALSE   FALSE
## 745      hockey 0.18750423 0.18208848 0.18241190   FALSE   FALSE   FALSE
## 791    football 0.35809946 0.32695992 0.32699761   FALSE   FALSE   FALSE
## 859      hockey 0.13365831 0.14645937 0.14683426   FALSE   FALSE   FALSE
## 1028   football 0.07579418 0.07783316 0.07790338   FALSE   FALSE   FALSE
## 1099   football 0.12718286 0.16433119 0.16441926   FALSE   FALSE   FALSE
## 1178   baseball 0.08432231 0.08863862 0.08854574   FALSE   FALSE   FALSE
## 1249 basketball 0.20762487 0.31444484 0.31412130   FALSE   FALSE   FALSE
## 1277 basketball 0.19064669 0.28213310 0.28183825   FALSE   FALSE   FALSE
## 1326     hockey 0.13142334 0.14326098 0.14363575   FALSE   FALSE   FALSE
## 1351   football 0.53386337 0.46802327 0.46803998    TRUE   FALSE   FALSE
## 1381   baseball 0.24734539 0.27430146 0.27428509   FALSE   FALSE   FALSE
## 1455   baseball 0.07421161 0.07309080 0.07300793   FALSE   FALSE   FALSE
```

```
## 1463   baseball 0.09939051 0.10875122 0.10863975   FALSE   FALSE   FALSE
## 1488   football 0.15716017 0.16960138 0.16960281   FALSE   FALSE   FALSE
## 1506     hockey 0.19701640 0.20900704 0.20925121   FALSE   FALSE   FALSE
## 1513     hockey 0.07474027 0.07368807 0.07393046   FALSE   FALSE   FALSE
## 1581   baseball 0.07208827 0.06955673 0.06947539   FALSE   FALSE   FALSE
## 1643 basketball 0.03000068 0.01877738 0.01857386   FALSE   FALSE   FALSE
## 1660   baseball 0.09593531 0.10602655 0.10597044   FALSE   FALSE   FALSE
## 1711   football 0.09650939 0.10401042 0.10395318   FALSE   FALSE   FALSE
## 1737 basketball 0.05236748 0.07523400 0.07495957   FALSE   FALSE   FALSE
## 1740     hockey 0.12431124 0.13723858 0.13764892   FALSE   FALSE   FALSE
## 1842     hockey 0.54944533 0.40352542 0.40392396    TRUE   FALSE   FALSE
## 1868   baseball 0.10701526 0.11887026 0.11878054   FALSE   FALSE   FALSE
## 1897   baseball 0.11021674 0.12224662 0.12214413   FALSE   FALSE   FALSE
## 1903     hockey 0.23506699 0.21437414 0.21468268   FALSE   FALSE   FALSE
## 1909   baseball 0.46354499 0.40640819 0.40641197   FALSE   FALSE   FALSE
## 1918   baseball 0.19855233 0.25243470 0.25244659   FALSE   FALSE   FALSE
```